PEMBUATAN APLIKASI EKSTRAKSI INFORMASI PADA WEB

ABSTRAKSI

Web merupakan tempat penyimpanan informasi yang terbesar. Ekstraksi informasi dari web telah dilakukan melalui berbagai penelitian, yang menghasilkan algoritma-algoritma (wrappers) yang mampu mengekstrak informasi, yang terstruktur secara sintaksis dan secara otomatis.

Dalam sebuah halaman web, informasi yang ditampilkan adalah dalam format HTML. tool yang digunakan untuk mengekstrak informasi dari HTML biasanya menggunakan sebuah modul yang disebut wrapper. Proses yang dilakukan dalam kegiatan wrapping meliputi menerima halaman web kemudian mengekstrak informasi dari halaman web, dan yang terakhir adalah menempatkan informasi yang telah diekstrak ke dalam bentuk XML.

Melalui penulisan ini akan dibuat dan uji coba sebuah aplikasi wrapper, yang dapat digunakan untuk mengekstrak informasi dari sebuah halaman web. Diharapkan dengan dilakukannya uji coba ini maka akan dapat memberikan kemudahan bagi user dalam mencari informasi yang diperlukan dari suatu halaman web.

Aplikasi wrapper ini dibuat dengan menggunakan Bahasa Pemograman Pythón 2.4.3 dan Boa Constructor v0.4.4 sebagai editornya.

PENDAHULUAN

Berbagai kegiatan saat ini, sangat membutuhkan beraneka ragam informasi, seperti untuk perencanaan, pengambilan keputusan, evaluasi dan sebagainya. Sumber informasi saat ini semakin banyak. Hal ini yang mendorong semakin memudahkan dalam pertukaran informasi.

Sebuah halaman *web* dapat menyajikan informasi dalam berbagai format, seperti gambar, data, suara, video. Kemudahan ini dapat berakibat sebuah informasi dapat berubah isinya atau koneksinya setiap waktu tanpa ada yang dapat mengaturnya. Setiap orang yang menginginkan sebuah informasi dapat dengan mudah mendapatkannya melalui media internet, walaupun informasi tersebut tidak sepenuhnya relevan dengan apa yang diinginkan *user*, oleh karena itu dibutuhkan sebuah metode pencarian informasi yang setidaknya mendekati atau menyaring informasi yang diinginkan oleh *user*. Permasalahan

yang timbul jika menggunakan metode manual adalah terkadang menyulitkan untuk memahami isi dari masing-masing halaman *web* tersebut dan membutuhkan waktu yang tidak sedikit.

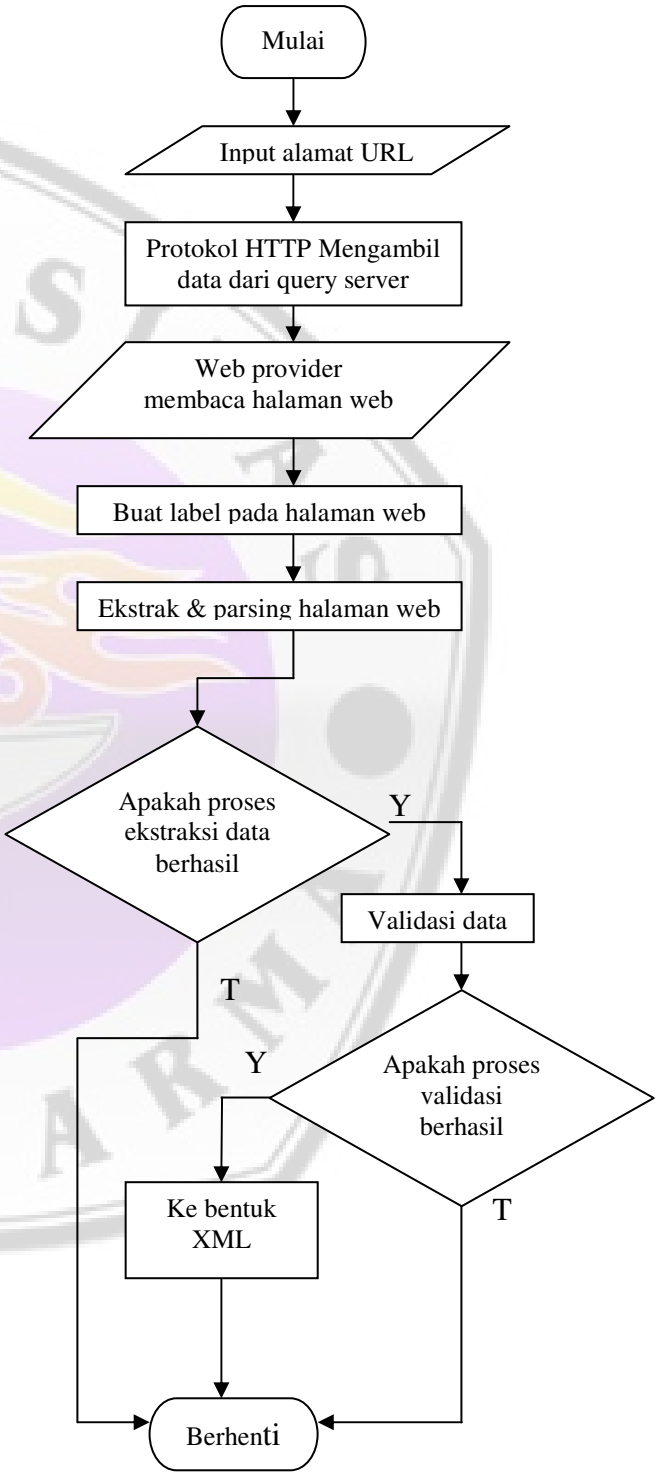
Aplikasi yang dibuat oleh penulis merupakan salah satu pilihan yang dapat memudahkan bagi user dalam pencarian informasi yang dibutuhkan dari suatu domain..

Proses yang dilakukan dalam kegiatan *extract* meliputi menerima halaman web kemudian mengekstrak informasi dari halaman web.

Dalam pembuatan aplikasi ini, penulis menggunakan bahasa pemprograman Python 2.4 dan Boa constructor v 0.4.4 sebagai editornya, dikarenakan Pythón *compatible* dengan windows dan Linux.

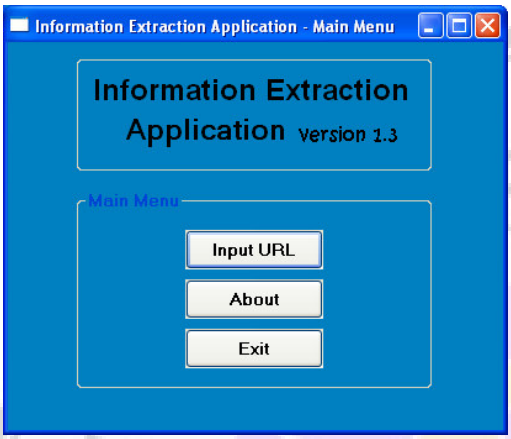
Sehingga penulis hendak menyajikan bahan penulisan ini dengan judul **"Pembuatan Aplikasi Ekstraksi Web dengan Menggunakan Phyton 2.4".**

FLOWCHART



TAMPILAN

Berikut ini adalah tampilan utama aplikasi wrapper :

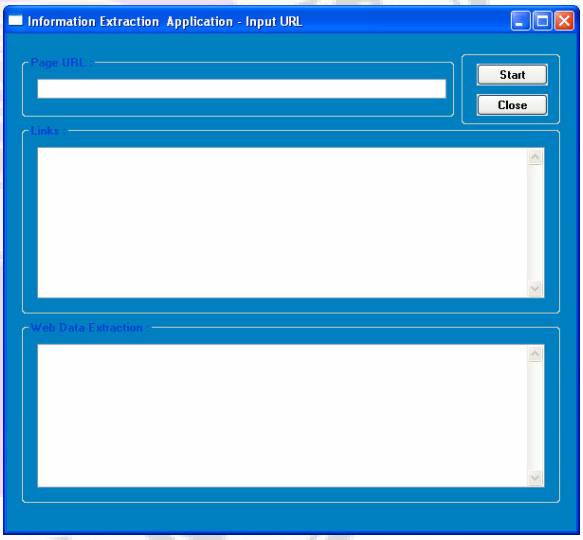


Berikut adalah penggalan program yang digunakan untuk menjalankan form utama pada aplikasi wrapper yang nantinya akan memanggil form input url dan form informasi tentang penulis :

```
    def __init__(self, parent):

self._init_ctrls(parent)

    def exit(self, event):
        self.Close(True)

    def InputURL(self, event):
        import InputURL

InputURL.create(self).Show(True
)

    def Tentang(self, event):
        import About

About.create(self).Show(True)
```

Dalam hal ini metode `def __init__` dengan paramenter `parent` digunakan untuk memanggil konstruktor class induk yaitu `class FrameMenu(wx.Frame)` yang merupakan class pada tampilan utama.

Berikut ini merupakan form Configuration Searching yang digunakan untuk memasukkan alamat url yang akan diekstrak halaman webnya :



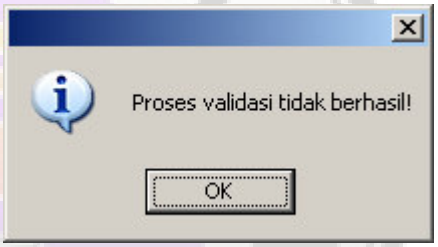
Ketika tombol start ditekan maka pada panel Links akan menampilkan output dari proses parsing halaman web apa saja yang terhubung dengan alamat URL yang dimasukkan pada panel

”page URL” dan panel ”web data extraction” akan menampilkan hasil ektraksi informasi yang terdapat pada halaman web tersebut.

```
def start(self, event):
      import urllib, sgmllib
      import xmlproc
      import xmlval
      import xmldtd

      # test web page
      d                              =
self.textCtrl1.GetValue()
      f = open(d)
      s = f.read()

      # process the web page
      myparser = MyParser()
      myparser.parse(s)

      self.textCtrl2.SetValue("
      %s"%(myparser.get_hyperli
      nks()))
      self.textCtrl3.SetValue("
      %s"%(p.get_application(My
      App())))
```

Dari penggalan program diatas terlihat mengimpor modul-modul dan fungsi-fungsi yang diperlukan untuk melakukan proses parsing dan ekstraksi informasi, dalam hal ini dideklarasikan juga variabel-variabel untuk menghubungkan textCtrl1 dengan panel links dan panel web data extraction yang dideklarasikan dengan `class MyParser(sgmllib.SGMLParser)` dan `class MyApp(xmlproc.Application).`

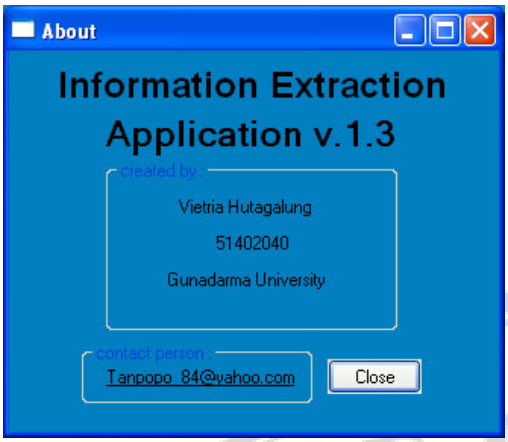
Apabila proses validasi tidak berhasil maka akan tampil MessageBox seperti dibawah ini :



Berikut ini adalah form informasi tentang penulis yang akan tampil jika menekan tombol About.

```
class About(wx.Frame):
    def     _init_ctrls(self,
prnt):
```

Kemudian program akan memanggil metode wx yang terdapat pada form utama. Berikut tampilan form tentang penulis :



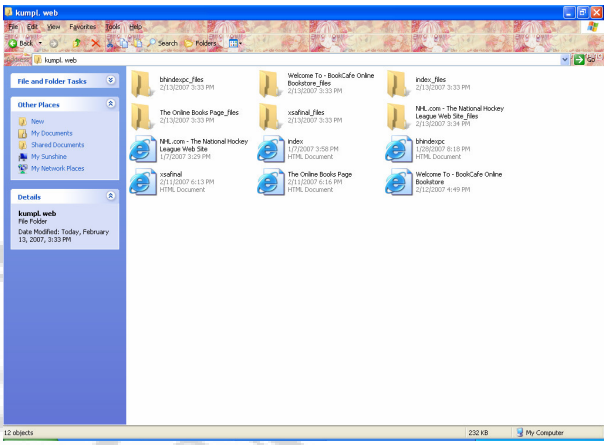
UJI COBA

Uji coba dapat dilakukan dengan syarat bahwa dalam komputer tersebut sudah terinstall program Python 2.4.3, Boa Constructor v0.4.4 dan PyXML-0.8.4.win32-py2.4.

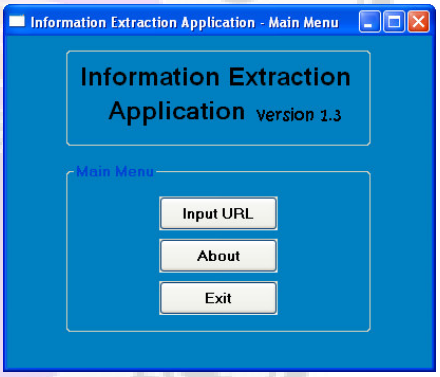
OFFLINE :

1. Masukkan program Wrapper kedalam directory C:\.

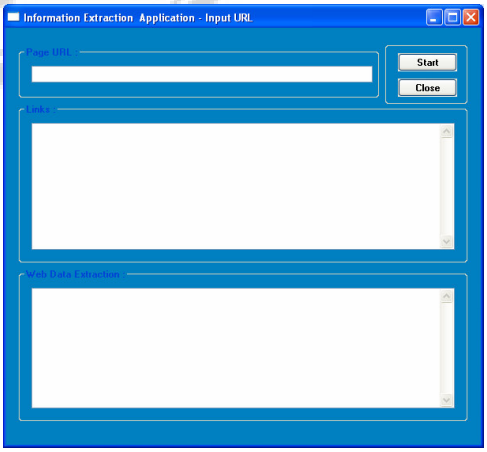
2. Masukkan halaman web, dalam hal ini NHL.com ke dalam directory D:\New\kumpl. web\



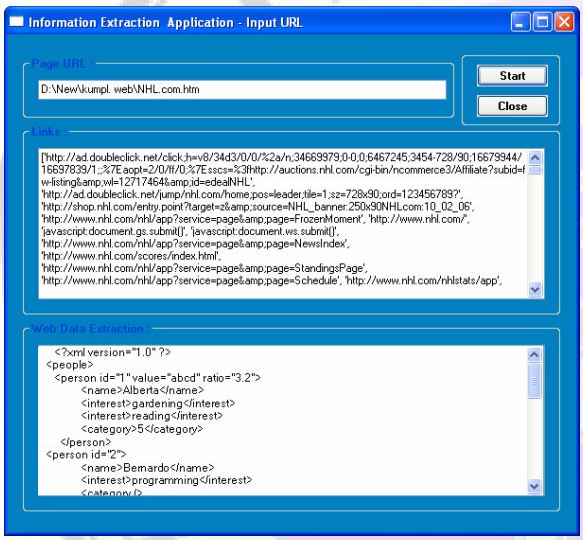
3. Buka file wrapper.py yang terdapat dalam folder program wrapper sehingga muncul tampilan sebagai berikut :



4. Kemudian tekan tombol Input URL maka muncul tampilan sebagai berikut :

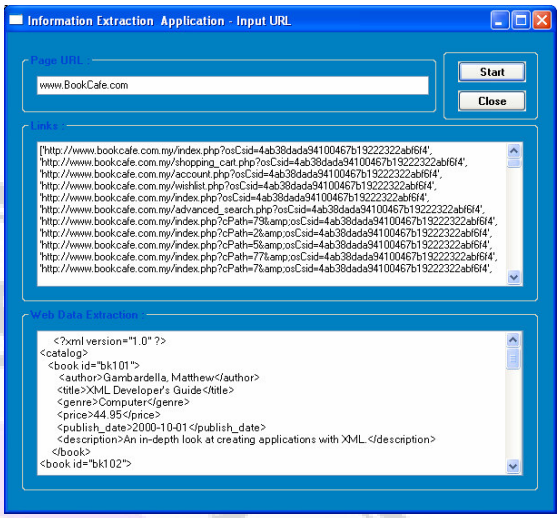


5. Input D:\New\kumpl. web\NHL.com.htm kedalam panel page URL kemudian tekan Start maka proses wrapper akan dilakukan, sehingga tampil hasil dari parsing dan ekstraksi informasi halaman tersebut.



ONLINE :

Lakukan uji coba secara online sama dengan langkah-langkah secara offline, hanya saja memasukkan alamat url secara langsung. Hasil proses wrapper online dengan domain www.bookcafe.com dapat dilihat pada gambar berikut:



KESIMPULAN

Web merupakan tempat penyimpanan informasi yang terbesar. Ekstraksi informasi dari web telah dilakukan melalui berbagai penelitian, yang menghasilkan algoritma-algoritma *(wrapper)* yang mampu mengekstrak informasi, yang terstruktur secara sintaksis dan secara otomatis.

Informasi yang ditampilkan dalam sebuah halaman web merupakan informasi yang tidak terstruktur atau yang semi terstruktur. Wrapper yang nantinya akan mengekstrak informasi yang tidak terstruktur atau semi terstruktur tersebut. Proses yang dilakukan dalam kegiatan *wrapping*

meliputi menerima halaman web kemudian mengekstrak informasi dari halaman web, dan yang terakhir adalah menempatkan informasi yang telah diekstrak ke bentuk XML. Hasil yang didapat dari proses *wrapping* biasanya berbentuk sebuah dokumen yang terstruktur seperti XML. Struktur XML itulah yang yang nantinya akan menjadi bahan informasi baru yang memudahkan user dalam memahami isi suatu halaman web.

Beberapa hal penting yang dapat mempengaruhi kinerja dari sebuah wrapper, antara lain:

1. Jumlah informasi yang tersedia dalam suatu halaman web.
2. Perubahan informasi dari suatu halaman web secara tiba-tiba yang menyebabkan gagalnya validasi informasi halaman web tersebut.
3. Besarnya bandwidth yang tersedia untuk sebuah wrapper.

Diharapkan untuk uji coba yang akan datang lebih memperhatikan ketiga faktor tersebut. Hal lain yang perlu diperhatikan juga yaitu spesifikasi komputer yang digunakan saat melakukan proses wrapper.

Daftar Pustaka

1. M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam and S. Slattery. *Learning to Extract Symbolic Knowledge from the World Wide Web*, Januari, 2002.

2. Ion Muslea, Steven Minton, Craig A. Knoblock, Kluwer, "Hierarchical Wrapper Induction for Semi-structured Information Sources", 1999.

3. Sidik, Betha, dkk, *Pemrograman Web Dengan HTML,* Informatika, Bandung, 2005.

4. Ramelan, Windiaprana, dkk, *Pengantar Internet,* Lembaga Pengembangan Komputerisasi, Universitas Gunadarma, 2000.

5. Noprianto, *Python dan Pemrograman Linux*, ANDI, Yogyakarta, 2002

6. Firar, Utdirartatmo, *Belajar Pemrograman WEB pada XML.* Yogyakarta, ANDI, 2003

7. http://www.python.org/

8. http://www.boost.org/libs/phyton

9. http://sundew.com